Integer programming (part 1)

Lecturer: Javier Peña

Convex Optimization 10-725/36-725

Coordinate descent

Assume $f(x) = g(x) + \sum_{i=1}^n h_i(x_i)$, with g convex, differentiable and each h_i convex.

To minimize f: start with some initial guess $x^{(0)}$, and repeat

$$
\begin{aligned}
x_1^{(k)} &\in \underset{x_1}{\mathrm{argmin}}\; f\Big(x_1, x_2^{(k-1)}, x_3^{(k-1)}, \ldots x_n^{(k-1)}\Big) \\
x_2^{(k)} &\in \underset{x_2}{\mathrm{argmin}}\; f\Big(x_1^{(k)}, x_2, x_3^{(k-1)}, \ldots x_n^{(k-1)}\Big) \\
x_3^{(k)} &\in \underset{x_2}{\mathrm{argmin}}\; f\Big(x_1^{(k)}, x_2^{(k)}, x_3, \ldots x_n^{(k-1)}\Big) \\
&\ldots \\
x_n^{(k)} &\in \underset{x_2}{\mathrm{argmin}}\; f\Big(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \ldots x_n\Big)
\end{aligned}
$$

for $k = 1, 2, 3, \ldots$.

Outline

Today:

- (Mixed) integer programming
- Examples
- Solution techniques:
  - relaxation
  - branch and bound
  - cutting planes

(Mixed) integer program

Optimization model where some variables are restricted to be integer

$$\begin{array}{rl} \min\limits_{x} & f(x) \\ \text{subject to} & x \in C \\ & x_j \in \mathbb{Z},\ j \in J \end{array}$$

where $f : \mathbb{R}^n \to \mathbb{R},\ C \subseteq \mathbb{R}^n$ and $J \subseteq \{1, \ldots, n\}$.

When $J = \{1, \ldots, n\}$, the above problem is a pure integer program.

Throughout our discussion assume f and C are convex.

Special case: binary variables

In some cases the variables of an integer program represent yes/no decisions or logical variables.

These kinds of decisions can be encoded via binary variables that take values 0 or 1.

Combinatorial optimization

A combinatorial optimization problem is a triple $(N, \mathcal{F}, c)$ where

- N is a finite ground set
- $\mathcal{F} \subseteq 2^N$ is a set of feasible solutions
- $c \in \mathbb{R}^N$ is a cost function

The goal is to solve

$$\min_{S \in \mathcal{F}} \sum_{i \in S} c_i$$

Many combinatorial optimization problems can be written as binary integer programs.

Knapsack problem

Determine the most valuable items to take in a limited volume knapsack.

$$
\begin{array}{rl}
\max\limits_{x} & c^{\mathsf{T}} x \\
\text{subject to} & a^{\mathsf{T}} x \leq b \\
& x_j \in \{0, 1\}, \; j = 1, \ldots, n
\end{array}
$$

here c_j and a_j are the value and volume of item j and b is the volume of the knapsack.

Assignment problem

There are n people available to carry n jobs. Each person can be assigned to exactly one job. There is a cost c_{ij} if person i is assigned to job j.

Find minimum cost assignment.

$$
\begin{aligned}
\min_x \quad & \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \\
& \sum_{i=1}^n x_{ij} = 1,\ j = 1, \ldots, n \\
& \sum_{j=1}^n x_{ij} = 1,\ i = 1, \ldots, n \\
& x_{ij} \in \{0, 1\},\ i = 1, \ldots, n,\ j = 1, \ldots, n.
\end{aligned}
$$

Facility location

There are $N = \{1, \ldots, n\}$ depots and $M = \{1, \ldots, m\}$ clients.

Fixed cost f_j associated to the use of depot j

Transportation cost c_{ij} if client i is served from depot j.

Determine what depots to open and what clients each depot serves to minimize the sum of fixed and transportation costs.

$$
\begin{aligned}
\min_{x,y} \quad & \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1,\ i = 1, \ldots, n \\
& x_{ij} \leq y_j,\ i = 1, \ldots, m,\ j = 1, \ldots, n \\
& x_{ij} \in \{0, 1\},\ i = 1, \ldots, m,\ j = 1, \ldots, n \\
& y_j \in \{0, 1\},\ j = 1, \ldots, n
\end{aligned}
$$

Facility location (alternative formulation)

Since all variables are binary, the mn constraints

$$x_{ij} \leq y_j, \; i = 1, \ldots, m, \; j = 1, \ldots, n$$

can be replaced by the n constraints

$$\sum_{i=1}^{m} x_{ij} \leq m y_j, \; j = 1, \ldots, n$$

Alternative formulation

$$\begin{aligned}
\min_{x,y} \quad & \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1, \; i = 1, \ldots, n \\
& \sum_{i=1}^{m} x_{ij} \leq m y_j, \; j = 1, \ldots, n \\
& x_{ij} \in \{0, 1\}, \; i = 1, \ldots, m, \; j = 1, \ldots, n \\
& y_j \in \{0, 1\}, \; j = 1, \ldots, n
\end{aligned}$$

K-means and K-medoids clustering

Assume $x^{(1)}, \ldots, x^{(n)} \in \mathbb{R}^d$.

K-means

Find partition $S_1 \cup \cdots \cup S_K = \{1, \ldots, n\}$ that minimizes

$$\sum_{i=1}^{K} \sum_{j \in S_i} \|x^{(j)} - \mu^{(i)}\|^2$$

where $\mu^{(i)} := \frac{1}{|S_i|} \sum_{j \in S_i} x^{(i)}$, centroid of cluster i.

K-medoids

Find partition $S_1 \cup \cdots \cup S_K = \{1, \ldots, n\}$ and select $y^{(i)} \in \{x^{(j)} : j \in S_i\}$, $i = 1, \ldots, K$ to minimize

$$\sum_{i=1}^{K} \sum_{j \in S_i} \|x^{(j)} - y^{(i)}\|^2$$

Best subset selection

Assume $X = \begin{bmatrix} x^1 & \cdots & x^p \end{bmatrix} \in \mathbb{R}^{n \times p}$ and $y \in \mathbb{R}^n$.

Best subset selection problem:

$$\begin{array}{rl} \min\limits_{\beta} & \frac{1}{2}\|y - X\beta\|_2^2 \\ \text{subject to} & \|\beta\|_0 \le k \end{array}$$

Here $\|\beta\|_0 :=$ number of nonzero entries of β.

Can you give an integer programming formulation to this problem?

Least median of squares regression

Assume $X = \begin{bmatrix} x^1 & \cdots & x^p \end{bmatrix} \in \mathbb{R}^{n \times p}$ and $y \in \mathbb{R}^n$.

Given $\beta \in \mathbb{R}^p$ let $r := y - X\beta$

Observe

- Least squares (LS): $\beta_{LS} := \underset{\beta}{\operatorname{argmin}} \sum_i r_i^2$
- Least absolute deviation (LAD): $\beta_{LAD} = \underset{\beta}{\operatorname{argmin}} \sum_i |r_i|$

Least Median of Squares (LMS)

$$\beta_{LMS} := \underset{\beta}{\operatorname{argmin}}(\text{median}|r_i|).$$

Can you give an integer programming formulation for LMS?

How hard is integer programming?

- Solving integer programs is much more difficult than solving convex optimization problems.
- Integer programming is NP-hard. There are no known polynomial-time algorithms for solving integer programs.
- Solving the associated convex relaxation (ignoring integrality constraints) results in an lower bound on the optimal value.
- The convex relaxation may only convey limited information:
  - Rounding to a feasible integer solution may be difficult
  - The optimal solution to the relaxation can be arbitrarily far away from the optimal solution to the integer program
  - Rounding may result in a solution far from optimal

Techniques for solving integer programs

Consider an integer program

$$z := \min_{x \in X} \; f(x)$$

(Assume X includes both convex and integrality constraints.)

Unlike convex optimization, there are no straightforward "optimality conditions" to verify that a feasible point $x^\star \in X$ is optimal.

A naive alternative: find a lower bound $\underline{z} \leq z$ and an upper bound $\overline{z} \geq z$ with $\underline{z} = \overline{z}$.

Techniques for solving integer programs

Consider an integer program

$$z := \min_{x \in X} \; f(x)$$

Algorithmic template

Find a decreasing sequence of upper bounds

$$\overline{z}_1 \geq \overline{z}_2 \geq \cdots \overline{z}_s \geq z$$

and an increasing sequence of lower bounds

$$\underline{z}_1 \leq \underline{z}_2 \leq \cdots \underline{z}_t \leq z$$

stop when $\overline{z}_s - \underline{z}_t \leq \epsilon$ for some specified tolerance $\epsilon > 0$.

Primal and dual bounds

How can we find upper and lower bounds for the problem

$$z := \min_{x \in X} \; f(x)$$

Primal bounds

Any feasible $x \in X$ yields an upper bound $f(x) \geq z$.
In some problems it is easy to find feasible solutions but this is not always the case.

Dual bounds

Finding lower bounds poses a different challenge. They are often called "dual" for reasons that will become apparent soon.
The most commonly used lower bounds are via relaxations.

Relaxations

We say that the problem

$$\min_{x \in Y} \; g(x)$$

is a relaxation of the problem

$$\min_{x \in X} \; f(x)$$

if

- $X \subseteq Y$
- $g(x) \leq f(x)$ for all $x \in X$

Observe that the optimal value of a relaxation is a lower bound on the optimal value of the original problem.

Convex relaxations

Consider the problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & x \in C \\ & x_j \in \mathbb{Z}, \ j \in J \end{aligned}$$

where $f : \mathbb{R}^n \to \mathbb{R}, \ C \subseteq \mathbb{R}^n$ are convex and $J \subseteq \{1, \ldots, n\}$.

Convex relaxation:

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & x \in C \end{aligned}$$

Lagrangian relaxations

Consider a problem of the form

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & Ax \le b \\ & x \in X \end{aligned}$$

If this problem is difficult, consider shifting some of the constraints to the objective: For $u \ge 0$ consider the Lagrangian relaxation

$$L(u) := \quad \begin{aligned} \min_x \quad & f(x) + u^{\mathsf{T}}(Ax - b) \\ \text{subject to} \quad & x \in X \end{aligned}$$

Observe that $L(u) \le z$ for all $u \ge 0$. The best (highest) such bound can be obtained by solving the dual problem

$$\max_{u \ge 0} \; L(u)$$

Observe that the dual is a concave maximization problem.

Lagrangian relaxation for facility location

Recall the facility location problem

$$\begin{aligned}\min_{x,y} \quad & \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij} x_{ij} \\ & \sum_{j=1}^{n} x_{ij} = 1,\ i = 1, \ldots, n \\ & x_{ij} \leq y_j,\ i = 1, \ldots, m,\ j = 1, \ldots, n \\ & x_{ij}, y_j \in \{0, 1\},\ i = 1, \ldots, m,\ j = 1, \ldots, n\end{aligned}$$

Lagrangian relaxation: for unrestricted v

$$\begin{aligned}L(v) := \min_{x,y} \quad & \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m}\sum_{j=1}^{n} (c_{ij} - v_i) x_{ij} + \sum_{i=1}^{m} v_i \\ & x_{ij} \leq y_j,\ i = 1, \ldots, m,\ j = 1, \ldots, n \\ & x_{ij}, y_j \in \{0, 1\},\ i = 1, \ldots, m,\ j = 1, \ldots, n\end{aligned}$$

Lagrangian relaxation for facility location

For each v, the Lagrangian relaxation $L(v)$ is easily solvable:

$$x_{ij}(v) = \begin{cases} 1 & \text{if } \; c_{ij} - v_i < 0 \text{ and } \sum_\ell (c_{\ell j} - v_\ell)^- + f_j < 0 \\ 0 & \text{otherwise} \end{cases}$$

$$y_j(v) = \begin{cases} 1 & \text{if } \; \sum_\ell (c_{\ell j} - v_\ell)^- + f_j < 0 \\ 0 & \text{otherwise.} \end{cases}$$

This gives both a lower bound $L(v)$ and a heuristic primal solution.

Furthermore, the subdifferential of $-L(v)$ is easy to compute.

Thus we can use a subgradient method to solve

$$\max_v \; L(v) \;\; \Leftrightarrow \;\; \min_v \; -L(v).$$

Branch and bound (B&B)

This is the most common algorithm for solving integer programs.

It is a divide and conquer approach.

Let $X = X_1 \cup X_2 \cup \cdots \cup X_k$ be a partition of X. Thus

$$\min_{x \in X} f(x) = \min_{i=1,\ldots,k} \{\min_{x \in X_i} f(x)\}.$$

Observe

- A feasible solution to any of the subproblems yields an upper bound $u(X)$ on the original problem.
- Key idea: obtain a lower bound $\ell(X_i)$ for each $\min_{x \in X_i} f(x)$.
- If $\ell(X_i) \geq u(X)$ then we do not need to consider $\min_{x \in X_i} f(x)$.

Branch and bound algorithm

Consider the problem

$$\begin{array}{rl} \min\limits_x & f(x) \\ \text{subject to} & x \in C \\ & x_j \in \mathbb{Z},\ j \in J \end{array} \tag{IP}$$

where $f : \mathbb{R}^n \to \mathbb{R},\ C \subseteq \mathbb{R}^n$ are convex and $J \subseteq \{1, \ldots, n\}$.

1. Solve the convex relaxation

$$\begin{array}{rl} \min\limits_x & f(x) \\ \text{subject to} & x \in C \end{array} \tag{CR}$$

2. (CR) infeasible $\Rightarrow$ (IP) is infeasible. Stop
3. Solution $x^\star$ to (CR) is (IP) feasible $\Rightarrow$ $x^\star$ solution to (P). Stop
4. Solution $x^\star$ to (CR) not (IP) feasible $\Rightarrow$ lower bound for (IP). Branch and recursively solve subproblems.

After branching

Key component of B&B

- After branching solve each of the subproblems.
- If a lower bound for a subproblem is larger than the current upper bound, no need to consider the subproblem.
- Most straightforward way to compute lower bounds is via a convex relaxation but other methods (e.g., Lagrangian relaxations) can also be used.

Tightness of relaxations

Suppose we have two equivalent formulations
(e.g., facility location)

$$\begin{array}{rl} \min\limits_{x} & f(x) \\ \text{subject to} & x \in C \\ & x_j \in \mathbb{Z},\ j \in J \end{array} \qquad \begin{array}{rl} \min\limits_{x} & f(x) \\ \text{subject to} & x \in C' \\ & x_j \in \mathbb{Z},\ j \in J \end{array}$$

with $C \subseteq C'$.

Which one should we prefer?

Convexification

Consider the special case of an integer program with linear objective

$$\begin{aligned} \min_x \quad & c^\mathsf{T} x \\ \text{subject to} \quad & x \in C \\ & x_j \in \mathbb{Z},\ j \in J \end{aligned}$$

This problem is equivalent to

$$\begin{aligned} \min_x \quad & c^\mathsf{T} x \\ \text{subject to} \quad & x \in S \end{aligned}$$

where $S := \mathsf{conv}\{x \in C : x_j \in \mathbb{Z},\ j \in J\}$.

Special case: integer linear programs

Consider the problem

$$\begin{array}{rl} \min\limits_{x} & c^{\mathsf{T}} x \\ \text{subject to} & Ax \le b \\ & x_j \in \mathbb{Z},\ j \in J \end{array}$$

Theorem

If A, b are rational, then the set

$$S := \textit{conv}\{x : Ax \le b,\ x_j \in \mathbb{Z},\ j \in J\}$$

is a polyhedron.

Thus the above integer linear program is equivalent to a linear program.

How hard could that be?

Cutting plane algorithm

We say that the inequality $\pi^{\mathsf{T}} x \leq \pi_0$ is valid for a set S if

$$\pi^{\mathsf{T}} x \leq \pi_0 \text{ for all } x \in S.$$

Consider the problem

$$\begin{array}{rl} \min\limits_{x} & c^{\mathsf{T}} x \\ \text{subject to} & x \in C \\ & x_j \in \mathbb{Z}, \ j \in J \end{array}$$

and let $S := \text{conv}\{x \in C : x_j \in \mathbb{Z}, \ j \in J\}$.

Cutting plane algorithm

Recall: $S = \text{conv}\{x \in C : x_j \in \mathbb{Z},\ j \in J\}$ and want to solve

$$\begin{array}{ll} \min\limits_{x} & c^\mathsf{T} x \\ \text{subject to} & x \in C \\ & x_j \in \mathbb{Z},\ j \in J \end{array} \qquad \text{(IP)}$$

Cutting plane algorithm

1. let $C_0 := C$ and compute $x^{(0)} := \text{argmin}\{c^\mathsf{T} x : x \in C_0\}$
2. for $k = 0, 1, \ldots$
   if $x^{(k)}$ is (IP) feasible then $x^{(k)}$ is an optimal solution. Stop
   else
      find a valid inequality (π, π_0) for S that cuts off $x^{(k)}$
      let $C_{k+1} := C_k \cap \{x : \pi^\mathsf{T} x \leq \pi_0\}$
      compute $x^{(k+1)} := \text{argmin}\{c^\mathsf{T} x : x \in C_{k+1}\}$
   end if
   end for

A valid inequality is also called a cutting plane or a cut

References and further reading

- Conforti, Cornuejols, and Zambelli (2014), "Integer programming"
- Wolsey (1998), "Integer programming"
- Belotti, Kirches, Leyffer, Linderoth, Luedke, and Mahajan (2012), "Mixed-integer nonlinear optimization"